REPORT

a.

```
student@systems-vm:~/os/operations_system/shell> python3 shell.py
/home/student/os/operations_system/shell $$$$ls
aa-chatbot.py  readfile.py  shell.py    testMinishell.shx
nw-chatbot.py  shell.docx    spinner.py
/home/student/os/operations_system/shell $$$$cat readfile.py
import sys

class FileReader:
    def __init__(self, filename):
        with open(filename) as f:
            content = f.readlines()
        self.file_lines = [x.strip() for x in content]

    def get_next_line(self):
        return self.file_lines.pop(0)


print('a')
if len(sys.argv) == 1:
    print('greater than 1')
if len(sys.argv) == 2:
    f = FileReader(sys.argv[1])
    c = 27
    while c>0:
        cv = f.get_next_line()
        if cv != "":
            if cv[0] != "#":
                cv = cv.split()
                print(cv[0])
        c-=1
print(sys.argv[1])
/home/student/os/operations_system/shell $$$$
```

```
/home/student/os/operations_system/shell $$$$exit
student@systems-vm:~/os/operations_system/shell>
```
b.

```
student@systems-vm:~/os/operations_system/shell> export PS1="elpaso"
elpasopython3 shell.py
elpasols
aa-chatbot.py  readfile.py  shell.py    testMinishell.shx
nw-chatbot.py  shell.docx    spinner.py
elpaso
```
c.

d.
```
elpasols &
elpasoaa-chatbot.py  readfile.py  shell.py      testMinishell.shx
nw-chatbot.py  shell.docx    spinner.py
```

```
student@systems-vm:~/os/operations_system/shell> python3 shell.py testMinishell.shx
/home/student/os/operations_system/shell $$$$ lsaa-chatbot.py  readfile.py  shell.py
testMinishell.shx
nw-chatbot.py  shell.docx    spinner.py
/home/student/os/operations_system/shell $$$$ chmod u+x spinner.py/home/student/os/o
perations_system/shell $$$$ mkdir -p mytestbin/home/student/os/operations_system/she
ll $$$$ cd mytestbin/home/student/os/operations_system/shell/mytestbin $$$$ echo imp
ort sys; print(f"echo {sys.argv[1]}") > line1.txt/home/student/os/operations_system/
shell/mytestbin $$$$ echo #!/usr/bin/env python3 > header.txt/home/student/os/operat
ions_system/shell/mytestbin $$$$ cat header.txt line1.txt > pyecho.py/home/student/o
s/operations_system/shell/mytestbin $$$$ chmod u+x pyecho.py/home/student/os/operati
ons_system/shell/mytestbin $$$$ cd ../home/student/os/operations_system/shell $$$$ e
cho pyecho.py > echo-out1.txt/home/student/os/operations_system/shell $$$$ echo pyec
ho.py > echo-out2.txt/home/student/os/operations_system/shell $$$$ nonExistentComman
d/home/student/os/operations_system/shell $$$$ ls nonExistentFilels: cannot access '
nonExistentFile': No such file or directory
/home/student/os/operations_system/shell $$$$ spinner.py 10000000 &/home/student/os/
operations_system/shell $$$$ wc < echo-out1.txt 1  1 10
/home/student/os/operations_system/shell $$$$ wc echo-out1.txt echo-out2.txt  | grep
 total/home/student/os/operations_system/shell $$$$ wc < echo-out1.txt 1  1 10
/home/student/os/operations_system/shell $$$$ wc echo-out1.txt echo-out2.txt  | grep
 total 2  2 20 total
 2  2 20 total
/home/student/os/operations_system/shell $$$$ ls nonExistentFilels: cannot access 'n
onExistentFile': No such file or directory
/home/student/os/operations_system/shell $$$$ spinner.py 10000000 &/home/student/os/
operations_system/shell $$$$ wc < echo-out1.txt 1  1 10
/home/student/os/operations_system/shell $$$$ wc echo-out1.txt echo-out2.txt  | grep
 total/home/student/os/operations_system/shell $$$$ wc < echo-out1.txt 1  1 10
/home/student/os/operations_system/shell $$$$ wc echo-out1.txt echo-out2.txt  | grep
 total 2  2 20 total
student@systems-vm:~/os/operations_system/shell>  2  2 20 total
```
e.

f.

```
/home/student/os/operations_system/shell $$$$python3 readfile.py testMinishell.shx
a
ls
chmod
mkdir
cd
echo
echo
cat
chmod
cd
echo
echo
nonExistentCommand
ls
spinner.py
wc
wc
exit
testMinishell.shx
/home/student/os/operations_system/shell $$$$
```

```
student@systems-vm:~/os/operations_system/shell> python3 shell.py
/home/student/os/operations_system/shell $$$$abc
Error: Command:'abc' not found./home/student/os/operations_system/shell $$$$
```
g.

i.

```
/home/student/os/operations_system/shell $$$$cd ..
/home/student/os/operations_system $$$$cd shell
/home/student/os/operations_system/shell $$$$
```

```
/home/student/os/operations_system/shell $$$$echo "abcde" > abc.txt
/home/student/os/operations_system/shell $$$$cat abc.txt
"abcde"
/home/student/os/operations_system/shell $$$$
```
j.

k.

```
/home/student/os/operations_system/shell $$$$wc < abc.txt
1 1 8
/home/student/os/operations_system/shell $$$$
```

```
/home/student/os/operations_system/shell $$$$cat readfile.py | sort
```
l.

```
        c-=1
    c = 27
class FileReader:
            content = f.readlines()
                cv = cv.split()
        cv = f.get_next_line()
    def get_next_line(self):
    def __init__(self, filename):
    f = FileReader(sys.argv[1])
        if cv != "":
            if cv[0] != "#":
if len(sys.argv) == 1:
if len(sys.argv) == 2:
import sys
print('a')
                print(cv[0])
    print('greater than 1')
print(sys.argv[1])
        return self.file_lines.pop(0)
        self.file_lines = [x.strip() for x in content]
    while c>0:
        with open(filename) as f:
student@systems-vm:~/os/operations_system/shell>
```

Ken Amamori
2/9/2021

This assignment was good introduction to handle input, file descriptor, fork since all of those components are in this lab and give me the chance to learn. First of all, understanding what each command means were difficult such as pipe, redirections and background run(&). Numeorus source and many turorial video to learn from the internet helped me a lot to get in track to proceed to complete this lab. Secondary, it was difficult to do every command manually since it requires what kind of operation is done in 'os' such as closing os.close(1). It was required to know what file descriptor is being used or should be closed in order to run correctly and know the order of calls that is needed.